Express Mail No. EL769181774 US

Date of Deposit: November 22, 2000

PATENT
0327.63678
99RSS429

ZERO-OVERHEAD SYMBOL RATE ADAPTATION SYSTEM FOR OVSF CODE

INVENTOR

Lih-Feng Tsaur

BACKGROUND OF THE INVENTION

1. Field of the Invention.

This invention relates to digital cellular communication systems, and more particularly, to digital cellular communication systems where a receiving station adapts to the symbol rate of the transmitting station without consuming overhead.

2. Related Art.

In cellular telephone systems, several conversations or other communications can take place simultaneously on a single carrier frequency. Interference between calls is avoided by multiplexing and/or encoding the signals for the various conversations, but cross-talk, in which one conversation spills into another conversation, is still a problem in such systems.

Orthogonal variable spreading factor (OVSF) codes have been adopted in third generation W-CDMA (wideband code division multiple access) wireless systems as channelization codes. One of the advantages is that the use of OVSF codes can reduce cross-talk, or multiple access interference (MAI), effectively.

Another advantage of OVSF codes is that OVSF codes can provide users with variable spreading length access and thus enable multi-rate adaptation. In other words, symbols having different lengths can be used at different times, depending on the circumstances. Multi-rate adaptation increases the capability of multimedia communication, which often has variable bit rate demand. Multi-rate adaptation also increases the ability to cope with time- and location-bearing conditions of the mobile, wireless channel environment.

Keeping the data transmission rate (chip rate) constant, a base station can use OVSF codes of different code lengths for transmitting symbols of different symbol duration. The received symbol's energy changes because energy is the product of power and the symbol duration (the OVSF code length in this case). MAI noise is unchanged because codes from different branches in the code tree are orthogonal irrespective of length. Therefor, the signal to noise ratio (SNR) or signal to information ratio (SIR) can be controlled effectively by applying an OVSF code of a particular length for a symbol, without suffering from MAI due to energy leakage from other channels.

This variable length characteristic provides stations the capability of increasing transmission efficiency by dynamically adjusting SNR or SIR and the symbol rate according to the channel environment and the QoS (quality of service) requirement of the application. However, the rate adaptation scheme must provide the receiver with a scheme for detecting the rate change.

Previous studies discussed rate adaptation using either rate information (RI) messages or blind rate detection. Using RI messages consumes wireless resources, incurring additional overhead for rate adaptation. Since resources of wireless communication are limited, it is desirable to reduce the overhead needed for rate detection at the receiver.

To improve performance and reduce this overhead, a blind rate detection scheme using a Viterbi-decoder has been suggested. The variable-rate data can be block-encoded with a cyclic redundancy checksum (CRC) and then convolutionally encoded before being transmitted. The receiver, with the knowledge of possible bit rates, uses a Viterbi-decoder to retrieve the convolutional-coded frame data and compare it with the CRC to keep the packet's integrity. The receiver must know the possible end bit positions $\{n_{end}\}$ of coded frame data, and the trellis path of the soft decision Viterbi-decoder should end up at the zero state at the correct end bit position. The blind rate detection scheme works with fixed frame systems, in which coded data is placed in frames with a fixed time interval. If the source data rate is lower than the maximum transmission rate, however,

only a partial frame is filled with data packets and the rest of the frame is empty (or idle). As a result, the transmission efficiency is not optimal due to the waste of slots in frames.

SUMMARY

This invention provides a communication system that transmits orthogonally encoded data at a symbol rate which is selected from a plurality of symbol rates. The data is encoded in a plurality of packets, each packet having a plurality of symbols having signal points in a field. The data is encoded by assigning a special set of OVSF codes (one code for each transmission rate) for each communication session. The special structure of the OVSF codes allows the receiver to detect changes in the symbol rate without overhead.

The codes are selected such that the signal points of consecutively transmitted symbols are correlated unless the rate is changed, and only particular orthogonal descendants of a selected maximum rate code are used for each communication session. The encoded data is transmitted to the receiver, and orthogonally decoded. A change in the rate of data transmission is recognized when the signal points of consecutively transmitted symbols are not correlated, or when symbol averaging produces an error symbol.

Other systems, methods, features and advantages of the invention will be or will become apparent to one with skill in the art upon examination of the following figures

and detailed description. It is intended that all such additional systems, methods, features and advantages be included within this description, be within the scope of the invention, and be protected by the accompanying claims.

BRIEF DESCRIPTION OF THE FIGURES

The components in the figures are not necessarily to scale, emphasis instead being placed upon illustrating the principals of the invention. Moreover, in the figures, like reference numerals designate corresponding parts throughout the different views.

FIG. 1 is a block diagram of a communication system using OVSF codes from which rate changes can be detected.

FIG. 2 is a diagram of signal points in an I-Q plane.

FIG. 3 is a more detailed block diagram of the transmitting system in a user station used in the communication system of Fig. 1.

FIG. 4 is a more detailed block diagram of the receiving system in a user station used in the communication system of FIG. 1.

FIG. 5 is a diagram of a tree structure showing the method of constructing OVSF codes from the Hadamard-Walsh sequence.

FIG. 6 is a diagram of a tree structure showing orthogonal and other branches.

FIG. 7 is a diagram showing a trellis code modulation structure, suitable for rate adaptation without consuming overhead.

FIG. 8 is another diagram of signal points in an I-Q plane used in the modulation structure of FIG. 7.

FIG. 9 is a diagram showing a reduction in symbol rate.

FIG. 10 is a diagram showing an increase in symbol rate.

FIG. 11 is a state diagram of a portion of an algorithm used to adapt the symbol rate.

FIG. 12 is a state diagram of another portion of an algorithm used to adapt the symbol rate.

FIG. 13 is a state diagram of another portion of an algorithm used to adapt the symbol rate.

Other systems, methods, features and advantages of the invention will be or will become apparent to one with skill in the art upon examination of the following figures and detailed description. It is intended that all such additional systems, methods, features and advantages be included within this description, be within the scope of the invention, and be protected by the accompanying claims.

DETAILED DESCRIPTION

In Fig. 1, a system overview is illustrated. A typical digital cellular telephone system includes at least one base station 100 and at least two user stations 102 which can

communicate with each other through the base station 100. The base station 100 and the user stations 102 each have a transmitter and receiver.

In one mode of operation, the base station 100 encodes analog signals such as voice communications or data communications digitally, for transmission to the user station 102. The digital signal can be encoded in any one of several OVSF codes $OVSF_1$, $OVSF_2$, $OVSF_3$. . .$OVSF_n$. The OVSF codes can be described mathematically as a series of signals in an I-Q plane, shown in Fig. 2. In Fig. 2, for example, the points 00, 10, 11 and 01 are orthogonal to each other because they do not have signals on the opposite points, i.e., A, B, C and D. These IQ symbols are encoded according to a selected OVSF code. With this encoding scheme, several communication sessions can be conducted simultaneously on a single carrier frequency.

The information is encoded and transmitted in packets of a predetermined length. The length of the packets determines the symbol rate, or rate at which packets are transmitted. Thus, packets having a short length are sent at a relatively high symbol rate, and longer packets are sent at lower symbol rates.

In this mode of operation, the user stations 102 decode the symbols Si sent from the base station 100, and use rate selection/detection logic 104 to determine which OVSF code was used to encode the data. In addition to sending regular communications to the base station, the user stations also send information regarding the signal to noise ratio and signal to information ratio (SNR/SIR) of data which has been received.

A control channel is used to initially select a family of OVSF codes for use in a communication session. Depending on the SNR/SIR of previously sent signals, the base station 100 selects the optimum OVSF coding for future transmissions, and changes the OVSF code more or less on the fly as needed. The user stations 102 constantly detect the symbol rate, to recognize changes immediately.

The system of Fig. 1 can be better understood with reference to Figs. 3 and 4. User data 300 is channel coded in a block coder 302, interleaved in an interleaver 304, processed in a scrambler 306 and a trellis coder 308, and mapped to corresponding signals on the I-Q plane in a mapper 310. The data 300 could include voice, video, data information or the like. Following the symbol mapper 310, each I and Q data symbol is encoded by an OVSF coder 312 to keep orthogonality between channels. Output OVSF code packets are further spread by a pseudo random noise (PN) circuit 314 to prevent MAI from other systems and adjacent cells in the cellular system.

Pilot symbols 316, which are separate from the data channel symbols, are separately coded by the longest OVSF code in another OVSF coder 318, and coded with the PN code in the PN circuit 314. These pilot symbols define the boundary between adjacent symbols, and are used for fast cell searching and OVSF coding synchronization. The real and imaginary parts of the output of the PN circuit 314 are modulated and added in a modulator 316 and converted to an analog signal in a digital to analog converter 318. The RF output of the converter 318 is transmitted through an antenna 320.

The MAI/QoS of the transmitted signals is detected by the receiving station and sent back to the transmitting station. The transmitting station has a receiver 322 and an MAI/QoS detector 324 which provide the MAI/QoS information to an $OVSF_n$ selector 326. The $OVSF_n$ selector 326 decides whether or not the OVSF code should be changed to produce a higher or lower symbol rate, and instructs the OVSF coder 318 accordingly.

At the receiving station, the RF signal is received in a receiver antenna 400 (Fig. 4). The signal r(x) includes a message signal m(t) and noise n(t), made up of MAI and Added White Gaussian Noise (AWGN), as indicated at 402. The real and imaginary parts of the signal are demodulated and correlated in real and imaginary component demodulators 404, 406, PN generators 408, 410, and integrators 412, 414, respectively, to produce a symbol S_i having an I component and a jQ component. The I and jQ components are added in an adder 416, and the I + jQ output (S_i) is multiplied by a feedback signal $C_{nq}(kq)$ in a multiplier 418 where C is an orthogonal code symbol having a rate q. The output of the multiplier 418 is correlated in an accumulator 420 to produce a $\rho_i(kq)$ output, where. The output of the circuit 420 is fed to a slicer and rate decision algorithm circuit 422, which feeds a portion of the $C_{nq}(kq)$ signal back to the multiplier 418. The output of the circuit 422 provides an input to a Viterbi decoder 424, de-scrambler 426, de-interleaver 428, and block decoder 430, which decode the signal and produce the desired voice or data output 432.

The receiving station returns current SNR and/or SIR status to the transmitter through the link's control channel. A multiple access interference (MAI)/quality of service (QoS) detector 434 determines SNR/SIR from the processor 422 and sends it to the transmitting station through the receiving station's transmitter 436. The transmitting station's transmitter decides the optimal symbol rate based on this SNR/SIR information.

OVSF codes can be constructed from the Hadamard-Walsh sequence using a tree structure method. According to the tree structure, there are 2^n codes at level n for notational purposes in Fig. 5, the subscript of the code symbol indicates the length of the code. Codes in the same level are numbered from 1 to 2^n as indicated in the parentheses. For instance, code $C_n(k)$ has length of n, and comes from the kth code from level $\log_2(n)$ of the tree, in which n is a power of 2.

A portion of such a tree structure is shown in Fig. 5. A code 500 has a length of 1, codes 502 and 504 have a length of 2, and codes 506, 508, 510 and 512 have a length of 4. Codes 514, 516, 518 and 520 have a length of n. These codes produce the symbols shown. Two of these codes' characteristics are noted here.

First, codes from different branches, $C_n(k)$, $C_m(k')$, such as code 504 and 508, are cyclic orthogonal to each other with the unit cyclic length I, (2 in C_2 (2)) equal to the greatest common divisor (GCD) of (n, m), where m is the length of another code. In other words, $C'_n(i)$, which is formed by rotating $C_n(i)$ by a multiple of I positions, is still

orthogonal to $C'_m(j)$, which is formed by rotation of $C_m(j)$ by a multiple of I positions. This property can be equivalently described as follows.

Without loss of generality it is assumed that m≥ n. Compare code 504, where n = 2, with code 508, where m = 4. Codes $C_n(k)$ and $C_m(k')$ can be expressed as $k' \notin \{ (k-1)m/n+1, (k-1)m/n+2, \ldots, km/n \}$ because they are in different branches. The cyclic orthogonality can be expressed as

$$\sum_{i=1}^{n} C_m^{i}(k) C_m^{i+jn}(k') = 0, \text{ for each of } j=0,1,\cdots,m/n-1$$

where $C_m^i(k)$ denotes the ith chip of code $C_n(k)$

Second, codes are not orthogonal if one is another's descendant. For instance, as seen in Fig. 6, $C_{64}(1)$ and $C_{128}(1)$ are not orthogonal, but $C_{64}(1)$ and $C_{128}(3)$ are orthogonal. This property can be expressed as below:

$$\sum_{i=1}^{n} C_m^{i}(k) C_m^{i+jn}(k') = \begin{cases} \neq 0, \text{ if } (k-1)\, m/n < k' \leq km/n \\ = 0 \text{ otherwise} \end{cases}$$

for each of j = 0,1, ···, m/n -1

Due to the second property, when an OVSF code is assigned to a mobile station 102, none of this code's descendants can be assigned to another mobile station 102 (Fig. 1) in the cell in order to keep orthogonality among users. The base station 100 assigns a whole branch of OVSF codes at a certain tree level to a mobile station 102 according to the channel's MAI and QoS requirement to meet constraints on data throughput, reliability, and delay. For non-realtime type applications with low bit error rate (BER) requirements such as File Transfer Protocol (FTP), a longer OVSF code (thus longer symbol duration) might be used to reduce the probability of re-transmission and therefor increase the overall system's efficiency. For real-time types of applications that can tolerate relatively high BER such as voice, a shorter OVSF code might be used to efficiently use the channel resources while keeping acceptable quality.

The length of the OVSF codes depends on the QoS requirement and channel MAI. The shortest OVSF code assigned to a mobile station 102 can be denoted as $C_n(k)$, k = 1,2...,n. Number n, which is a power of 2, corresponds to the maximum symbol rate that can be used by the mobile station 102 assigned with this code. Once the base station 100 grants this OVSF code $C_n(k)$, none of its descendent code can be assigned to or used by another mobile user (to make codes of all users orthogonal). Thus, the station assigned with code $C_n(k)$ may be able to use codes that are descendants of $C_n(k)$ in the code tree

in order to change the symbol rate dynamically. The receiver must be able to detect the change of the symbol rate.

A simple rate detection algorithm for the use in the present invention will now be described. This algorithm engages "signal symbol averaging" on adjacent I-Q symbols. For this algorithm to function properly, there are two major requirements.

First, the signal points of consecutively transmitted symbols, S_t and S_{t+1}, must not be opposite on the constellation I-Q diagram, i.e. the symbol S_t is not equal to $-S_{t+1}$ when no rate change is made. In other words, the summation of these two signal points does not equal zero. This kind of symbol stream can be designed either using an asymmetric signal set or using convolution codes, as will be described.

A simple way to avoid consecutively transmitted symbols being opposite is to have a signal set without opposite signal pairs on the I-Q plane, i.e., $\forall$ (a,b), if $a+bi \in S \Rightarrow -a-bi \notin S$, where S is the signal set. This signal set S can be produced by allocating each signal point on the I-Q plane asymmetrically.

Trellis-coded modulation (TCM) can be designed to prevent the consecutively transmitted I-Q symbols from being opposite to each other. Fig. 7 illustrates a simple trellis code. The encoder, which uses sequential logic with constraint length K=3, accepts one input-bit and produces three output-bits. The generator vectors of the sequential logic are g1=[100], g2=[100], and g3=[111] respectively.

The corresponding 8-PSK signal constellation mapping is shown in Fig. 8. This trellis code, for example, prevents the consecutively transmitted I-Q symbols from being opposite to each other. If, for example, the state 000 is detected at 700, the next signal could be either 000, or 111. Referring to Fig. 8, the point opposite to 000 is 101, which is not available in the trellis of Fig. 7. Thus this opposite point cannot be reached. A careful design can simultaneously achieve low symbol error probability, as well.

Second, only particular descendants of the maximum-rate code, e.g. $C_n(k)$, can be used for rate adaptation. This algorithm requires that the transmitter and receiver pair use $C_{nq}(kq)$ for symbol rate q times slower than the highest symbol rate granted, where q can only be powers of 2. In other words, as an OVSF code with code length nq, code $C_{nq}(kq)$ in the code tree must be used.

The receiving station performs symbol decoding by applying the OVSF code associated with the currently estimated symbol duration to each segment of the received signal estimated to be a symbol. To detect a transmitter reducing symbol rate (Fig. 9), i.e., that the transmitter intends to reduce or has reduced the symbol rate, the algorithm works as follows. Assume the current symbol rate is q times slower than the maximal symbol rate denoted as R_{max}, i.e. $R_{current} = R_{max}/q$. If the transmitter decides to reduce S_i's symbol rate R_i to R_{max}/p, it updates its code to $C_{np}(kp)$, which is generated by concatenating codes $C_{nq}(kq)$ and $-C_{nq}(kq)$ recursively. In other words, the transmitter multiplies symbol S_i with OVSF code $C_{np}(kp)$. In this case p is larger than q. The

receiver at first still applies its current code $C_{nq}(kq)$ to the first nq OVSF chips of symbol S_i and decodes out what the receiver perceives as a symbol. Note that the transmitter has made the duration of S_i longer than nq chips, so the actual symbol S_i has not finished its duration at this point.

When the receiver continuously uses $C_{nq}(kq)$ to decode the next nq OVSF chips' intervals of S_i, what is perceived as the next symbol S_{i+1} by the receiver is an opposite point of what has been perceived as S_i 902 by the receiver on the I-Q plane. This is due to the structure of the OVSF code used by the transmitter when expanding the symbol duration.

Opposite polarity of consecutively transmitted symbols contradicts design, i.e., signal points of adjacent symbols are never opposite points in the I-Q plane. Thus, when this event happens, the receiver recognizes that it should update the OVSF code to the longer OVSF code $C_{2nq}(2kq) = [C_{nq}(kq) \mid -C_{nq}(kq)]$. The receiver then uses this new OVSF code sequence for the next symbol decoding and iterates the previous operations until what are considered as the next two symbols are not opposite on the I-Q plane.

If the transmitter increases the symbol rate (Fig. 10), it reduces its OVSF code length to np, where p<q. In this case the transmitter uses code $C_{np}(kp)$, and the receiver still uses code $C_{nq}(kq)$, which is constructed recursively from $[C_{np}(kp) \mid -C_{np}(kp)]$. What is perceived as an output symbol, ρ, by the receiver with the duration of nq OVSF chips is in fact a sequence of multiple symbols transmitted by the transmitter. In this case ρ is

not close enough to any signal points of the constellation in the I-Q plane to be considered a valid symbol.

The average of these $q/p = 2^m$ symbols produces an error symbol, although the consecutive symbols actually transmitted are never the opposite in the I-Q plane (symbol error). Also, the symbol constellation is designed such that averaging multiple signal points results in symbol error. In this case, the receiver reduces the estimated symbol duration by half and tries the corresponding OVSF code. The receiver performs this repeatedly until the estimated symbol duration and the corresponding OVSF code decodes a valid symbol. When the receiver has decoded a valid symbol with $C_{nq}(kq')$ (e.g., code 512 in Fig. 5), the receiver could further decode two halves of the corresponding signal portion with $C_{nq'/2}(kq/2')$ (code 504 in Fig. 5) each for robustness (such as better SNR/SIR). Then, these two nq'/2-chip outputs, denoted by ρ and ρ', must be opposite in the I-Q plane if the symbol rate is unchanged and should stop at C_{nq} (kq') (code 512).

A more formal description of the rate detection algorithm is shown in Figs. 11, 12 and 13. Each iteration starts at one of the four states: (G_t,G_{t+1}), (G_t,B_{t+1}), (B_t,G_{t+1}), (B_t,B_{t+1}), where G is a valid (good) symbol and B is an invalid (bad)symbol, and the t +1 symbol follows the t symbol in time. These are the four combinations of the states describing whether a perceived symbol is close enough to a signal constellation in the I-Q plane to be valid. Variable q indicates the current number of OVSF chips that the

receiver tries as the length of the symbol; namely, the current symbol length is nq OVSF chips.

S_i (Fig. 4) denotes the portion of the received signal that the receiver considers as the *i*th symbol transmitted. Variable q at each iteration means that the algorithm is considering the symbol duration of nq OVSF chips. If the algorithm is considering S_t, then the length of S_t is nq OVSF chips for the current value q. Note that as q is updated, the portion of the signal tried by the receiver changes as the symbols S_t change. The normalized correlation ρ_t between S_t and the OVSF code corresponds to the current value of q at the iteration. F_t is defined as a binary variable flag {G,B} indicating whether ρ_t is a valid signal point of the constellation in the I-Q plane. F_t=G means that ρ_t is a valid signal point. In Figs. 11, 12, and 13, for simple notation G_t, B_{t+1} denotes that a first signal is good and the signal following it in time is bad.

Referring again to Fig. 11, assuming an initial state 900 of G_t, G_t+1, the algorithm determines whether $\rho_t = -\rho_t + 1$ at step 1102. If so, the algorithm uses a longer code word Q, such as 2Q, and recomputes ρ_t +1. The output of the correlator $\rho_t + \rho_{t+1}$ becomes ρ_t, and a new ρ_{t+1} is computed at step 1104. The flags F_t and F_{t+1} are checked at 1106, and the result determines whether the algorithm returns to step 1100 (when both flags are good) or to step 1200 (Fig. 12), 1300 (Fig. 13) or 1302.

If ρ_t does not equal $-\rho'_{t+1}$ in step 902, then ρ_t is good, and it is output at 1108. The flags F_{t+1}, F_{t+2} are checked at step 1110, and the result of step 1110 determines whether the algorithm returns to step 1100, 1200, 1300 or 1302.

If the system determines that the two symbols are in state 2 (G_t, B_{t+1}), the algorithm follows the steps shown in Fig. 12. The first symbol ρ_t is output in step 1202. In step 1204, q is reduced by 1/2, and ρ_{t+1} and ρ_{t+2} are computed. The state is checked again in step 1206, and the result is used to determine whether the system returns to step 1100, 1200, 1300 or 1302.

The algorithm for states 3 and 4 (1300, 1302) is shown in Fig. 13. Since the first symbol is bad in both states, q is reduced by a factor such as 1/2, and ρ_t and ρ_{t+1} are computed in step 1304. The states are checked again in step 1306 to determine whether the system should return to step 1100, 1200, 1300 or 1302.

Thus, when referring again to Fig. 11, if both the symbols are considered good, the first symbol (ρ_t) is output and the next two symbols are checked in step 1110. Thus, when there is no change in rate, the system goes from step 1100 to 1102, 1108 and 1110, and returns to step 1100. If the rate changes, though, the change is detected in step 1102, and the system goes to step 1104 and increases the code length q. The symbols ρ_t and ρ_{t+1} are combined to create a new ρ_t , and a new ρ_{t+1}, which follows the new ρ_t, is computed in step 1104. If the flags are found to satisfy state 1, then the receiver has found a correct symbol rate and the system returns to step 1100. If the symbol length has

not been increased enough, both symbols will still be checked as good, even if adjacent symbols are not opposite, and the system will return again to step 1100. If the symbols are not opposite, that will be detected again in 1102, and the symbol length will be doubled again. That process will be repeated until adjacent points are not opposite.

Increasing the length of the code word q in step 1104 suggests that the symbol rate has been decreased. If in fact the symbol rate has been increased, the result of step 1106 will be either state 2, state 3 or state 4.

If the rate changed again immediately, then state 2 would be detected. If after increasing the length q it is determined that the initial first symbol ρ_t is bad, then states 3 or 4 would be detected.

In Fig. 11, the first assumption is that the symbol rate has been reduced. If in fact it has been increased, the step 1106 will generate state 3 or state 4, and q will be reduced. Since the second symbol is bad in state 2, the first symbol is output at step 1202 and the system processes on the second symbol ρ_{t+1}. If the first symbol is bad, as in steps 1300 and 1302, the system recomputes the first symbol and continues.

Referring again to Fig. 11, when the system is in state 1 and step 1102 produces a NO result, the first symbol ρ_t is output, and the next symbol is checked in step 1110. The next symbol is denoted ρ_{t+2} in step 1110. If G_{t+2} is good, the system returns to step 1100. If the symbol ρ_{t+2} is bad, step 1200 in Fig. 12 is entered.

In Fig. 12, the output ρ_t is identified in step 1202. In step 1204, q is divided by 2 and ρ_{t+1} and ρ_{t+2} are computed. The new outputs ρ_{t+1} and ρ_{t+2} are checked in step 1206 and resulting flags F_{t+1} and F_{t+2} determine whether the algorithm enters state 1, state 2, state 3 or state 4.

In Fig. 13, the first flag F indicates a bad output in both state 3 (1300) and state 4 (1302). In step 1304, q is divided by 2 and ρ_t and ρ_{t+1} are computed. The outputs ρ_t and ρ_{t+1} are checked at step 1306, and the resulting flags F_t, F_{t+1}, determine whether the algorithm enters state 1, state 2, state 3 or state 4.

While the various embodiments of the application have been described, it will be apparent to those of ordinary skill in the art that many more embodiments and implementations are possible that are within the scope of this invention. Accordingly, the invention is not to be restricted except in light of the attached claims and their equivalents.